

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Lise Wiseman et al.                      Art Unit : 2194  
Serial No. : 09/927,957                                  Examiner : Li Zhen  
Filed : August 9, 2001  
Title : INTEGRATING ENTERPRISE SUPPORT SYSTEMS

**Mail Stop Appeal Brief - Patents**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**BRIEF ON APPEAL**

**(1) Real Party in Interest**

The real party in interest is Accenture Global Services GmbH, a Swiss corporation having a place of business at Herrenacker 15, CH-8200 Schaffhausen, Switzerland, as evidenced by an assignment executed April 1, 2003 and submitted for recordation at the U.S. Patent Office on April 3, 2004. The assignment was recorded at reel 013929 frame 0176 on April 8, 2003.

**(2) Related Appeals and Interferences**

Neither Appellant, nor Appellant's legal representative, nor the assignee are aware of any appeals or interferences that will directly affect or be affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

Claims 1-48 are pending and on appeal.

**(4) Status of Amendments**

No amendments were made after the final office action.

**(5) Summary of Claimed Subject Matter**

## Claim 1

<i>A computer-implemented method of exchanging information among applications "</i>	Exemplary applications shown in FIG. 3 are the billing application <b>302</b> , the provisioning application <b>304</b> , the customer relation application <b>406</b> , and the billing application <b>308</b> in FIG. 3.
<i>providing a plurality of transformers, each transformer corresponding to a unique transformation from one format into another</i>	The "transformers" are discussed in paragraph 32 on page 11, and in paragraphs 43-44 on page 19. FIG. 3 shows the logical placement of a transformer <b>320</b> between the subscriber <b>318</b> and driver <b>322</b> on the lower left side. The formats referred to are the various enterprise application formats or vendor-specific formats associated with each application, as discussed in paragraph 32, as well as the IDL (Interface Data Language), which is a common format discussed in paragraph 31.
<i>using a first transformer to transform a data object from a format understandable by a first application into a common format data object;</i>	The "data object" that is to be translated is a business event. See paragraph 29 and also paragraph 31. The "first transformer" is any transformer in FIG. 3 that separates a publisher <b>317</b> from a driver <b>316</b> . The "format understandable by a first application" is the vendor-specific format discussed above. The "common format data object" is a data object after translation into the IDL, or common format discussed in paragraph 31.
<i>publishing the common format data object to a selected communication channel, the channel being selected on the basis of the data object;</i>	A "communication channel" is a connector. Connectors <b>316</b> , <b>317</b> are shown in FIG. 3 and discussed in paragraphs 28, 41-44, and 47-48. The selection of a channel (i.e. a connector) on the basis of the data object (i.e. the business event) is discussed in connection with the operation of the channel architecture <b>314</b> in paragraph 17 and 40. The act of publishing is referred to in paragraph 42. The role of the publisher <b>317</b> in publishing is described in paragraph 41. A specific example of a publication is described in paragraph 47. That the data object is published in the common format is suggested by FIG. 3, which shows the logical position of the publisher <b>317</b> as being on the other side of the transformer <b>320</b> from the driver <b>322</b> .
<i>subscribing to the communication channel to retrieve the published common format data object; and</i>	An example of an application subscribing to a channel is described in paragraph 48. The role of the subscriber <b>318</b> is discussed in paragraph 41. That the data object retrieved is one published in a common format is suggested by a logical position of the subscriber <b>318</b> in FIG. 3, which is separated from the driver by the transformer <b>320</b> , that translates between the two formats.
<i>using a second transformer to transform the common format data object into a format understandable by a second application."</i>	The "second transformer" is any transformer in FIG. 3 that separates a subscriber <b>318</b> from a driver <b>316</b> . The act of subscribing is described in paragraph 42.

### Claim 18

<i>A computer-implemented method of facilitating the exchange of information among applications, the method comprising:</i>	Exemplary "applications" are described in connection with claim 1.
<i>receiving a data object from a first application;</i>	Data objects are described in connection with claim 1. The act of "receiving a data object" is referred to in paragraph 31 and 34.
<i>using a first controller to route the received data object to a first transformer;</i>	A "controller" is described in paragraph 33. Its function is given as routing a business event to the correct transformer.
<i>using the first transformer to transform the data object from a first format used by the first application into a common format object;</i>	Please see the discussion of the corresponding step in claim 1.
<i>publishing the common format object to a communication channel;</i>	Please see the discussion of the corresponding step in claim 1.
<i>receiving a request from a subscribing application to subscribe to the communication channel;</i>	Please see the discussion of the "subscribing" step in claim 1. Paragraph 31 describes applications as issuing requests.
<i>using a second controller to route the common format object to a second transformer;</i>	Controllers are discussed above in connection with the step of "using a first controller."
<i>using the second transformer to transform the common format object into a data object in a second format used by the subscribing application; and</i>	Please see the discussion of the corresponding step in claim 1.
<i>sending the data object in the second format to the subscribing application.</i>	Please see the discussion of the "subscribing" step in claim 1.

### Claim 28

<i>A system for facilitating the exchange of information among applications, the system comprising:</i>	Please see the discussion of the preamble of claim 1.
<i>a plurality of digital computers, each of which executes an application, each application being configured to exchange information representative of business events with other applications; and</i>	Four such computers are shown in FIG. 3.
<i>an integration hub in data communication with each of the digital computers for enabling transfer of information representative of</i>	The integration hub 300 is shown in FIG. 3.

<i>business events between applications, the integration hub including a computer-readable medium on which encoded instructions for causing a computer to define</i>	
<i>plurality of process models each defining one or more conditions for sending a business event from an application to one or more other applications;</i>	Process models are described in paragraph 30.
<i>a shared object model configured to store data objects received from applications in a common format;</i>	A shared object model 312 is shown in FIG. 3 and described in paragraph 31.
<i>a plurality of transformer classes configured to translate data object from a format used by one or more applications into the common format or vice versa; and</i>	Transformers, a.k.a. transformer classes, are described in paragraph 32 and labeled 320 in FIG. 3.
<i>a plurality of controller classes configured to route data objects to associated transformer classes.</i>	Controller classes are described in paragraph 33.

### **Claim 38**

Please see the discussion of the corresponding steps in claim 18.

### **(6) Grounds of Rejection**

Claims 38-48 stand rejected under 35 USC 101.

Claims 1-11, 16-17 stand rejected under 35 USC 103 as being rendered obvious by the combination of *Gupta* and *Koo*.

Claims 12-15 and 18-48 stand rejected under 35 USC 103 as being rendered obvious by the combination of *Gupta*, *Koo*, and *Bass*.

### **(7) Argument**

#### ***1. Law on Obviousness***

"It is well established that the burden is on the PTO to establish a prima facie showing of obviousness, *In re Fritsch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (C.C.P.A., 1972)."

"It is well established that there must be some logical reason apparent from the evidence or record to justify combination or modification of references. *In re Regal*, 526 F.2d 1399 188,

U.S.P.Q.2d 136 (C.C.P.A. 1975). In addition, even if all of the elements of claims are disclosed in various prior art references, the claimed invention taken as a whole cannot be said to be obvious without some reason given in the prior art why one of ordinary skill in the art would have been prompted to combine the teachings of the references to arrive at the claimed invention. *Id.* Even if the cited references show the various elements suggested by the Examiner in order to support a conclusion that it would have been obvious to combine the cited references, the references must either expressly or impliedly suggest the claimed combination or the Examiner must present a convincing line of reasoning as to why one skilled in the art would have found the claimed invention obvious in light of the teachings of the references. *Ex Parte Clapp*, 227 U.S.P.Q.2d 972, 973 (Board. Pat. App. & Inf. 1985)."

"The mere fact that the prior art could be so modified would not have made the modification obvious unless the prior art suggested the desirability of the modification." *In re Gordon*, 221 U.S.P.Q. 1125, 1127 (Fed. Cir. 1984).

Although the Commissioner suggests that [the structure in the primary prior art reference] could readily be modified to form the [claimed] structure, "[t]he mere fact that the prior art could be so modified would not have made the modification obvious unless the prior art suggested the desirability of the modification." *In re Laskowski*, 10 U.S.P.Q. 2d 1397, 1398 (Fed. Cir. 1989).

"The claimed invention must be considered as a whole, and the question is whether there is something in the prior art as a whole to suggest the desirability, and thus the obviousness, of making the combination." *Lindemann Maschinenfabrik GMBH v. American Hoist & Derrick*, 221 U.S.P.Q. 481, 488 (Fed. Cir. 1984).

Obviousness cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion supporting the combination. Under Section 103, teachings of references can be combined only if there is some suggestion or incentive to do so. *ACS Hospital Systems, Inc. v. Montefiore Hospital*, 221 U.S.P.Q. 929, 933 (Fed. Cir. 1984) (emphasis in original, footnotes omitted).

“The critical inquiry is whether ‘there is something in the prior art as a whole to suggest the desirability, and thus the obviousness, of making the combination.’” *Fromson v. Advance Offset Plate, Inc.*, 225 U.S.P.Q. 26, 31 (Fed. Cir. 1985).

## **2. Law on tangible subject matter**

According to MPEP 2106 B(1)(a), a “computer-readable medium encoded with a computer program is a computer element which defines structural and functional interrelationships between the computer program and the rest of the computer which permit the computer program’s functionality to be realized, and is thus statutory.”

Similarly, in *In re Beauregard*, 53 F.3d 1583 (Fed. Cir. 1995), the court stated the Commissioner’s agreement that “computer programs embodied in a tangible medium... are patentable subject matter under 35 USC 101.”

## **3. The combination of *Gupta* and *Koo* fails to render obvious claims 1-11, 16-17**

Independent claim 1, and its dependent claims 2-11 and 16-17 are argued as a group.

To facilitate discussion, independent claim 1 is reproduced below with numbered paragraphs:

*1. A computer-implemented method of exchanging information among applications, the method comprising:*

*[1] providing a plurality of transformers, each transformer corresponding to a unique transformation from one format into another;*

*[2] using a first transformer to transform a data object from a format understandable by a first application into a common format data object;*

*[3] publishing the common format data object to a selected communication channel, the channel being selected on the basis of the data object;*

*[4] subscribing to the communication channel to retrieve the published common format data object; and*

*[5] using a second transformer to transform the common format data object into a format understandable by a second application.*

Under the following four sub-headings, Appellant presents four distinct reasons for urging patentability of claim 1. Subheadings 3.1 and 3.3 address whether or not the references, even when combined, yield the claimed subject matter. Subheadings 3.3 and 3.4 address the propriety of combining the references.

**3.1 Gupta discloses publishing and subscribing to events not channels**

*Gupta* discloses publishing and subscribing to “events,” not “channels.” Events and channels are different. A “communication channel” suggests a channel, or pathway for providing communication between two entities; an “event” suggests data representing something that occurs.

The Examiner considers step [4] of claim 1 (“subscribing to the communication channel”) to be disclosed by step 102 of FIG. 2 in *Gupta*. That figure expressly refers to an “application collaboration module” 40 as subscribing to an “event.” Based on this, Appellant infers that the Examiner regards “communication channel” and “event” as synonyms.

The Examiner further considers step [3] of claim 1 (“publishing the common format data object to a selected communication channel”) to be disclosed by step 105 in FIG. 2, which refers to an “interchange end” (of a connector 30) as publishing an “event.” Based on this, Appellant infers that the Examiner regards “data object” and “event” as synonyms.

Apparently, the Examiner:

1. regards a *Gupta* “event” as corresponding to claim 1’s “communication channel”; and
2. regards a *Gupta* “event” as *also* corresponding to claim 1’s “data object”.

Therefore, by transitivity, it follows logically that the Examiner regards claim 1’s “communication channel” to be the same as claim 1’s “data object”.

By in effect assigning the *Gupta* “event” two different roles to play in claim 1, the Examiner wreaks havoc on any attempt to reasonably interpret claim 1. For example, claim 1’s limitation

“publishing the common format data object to a selected communication channel, the channel being selected on the basis of the data object;”

becomes almost tautological:

“publishing the common format *event* to a selected *event*, the *event* being selected on the basis of the *event*,”

Similarly, claim 1's limitation of

subscribing to the communication channel to retrieve the published common format data object;

likewise becomes tautological:

subscribing to the *event* to retrieve the published event;

Subscribing to a “channel” clearly differs from subscribing to an “event.” When one subscribes to a “channel,” one receives *all* events placed on that channel regardless of the type of event. When one subscribes to an “event,” one receives only those events subscribed to, regardless of what other events might be on a channel. By assigning *Gupta*'s “event” to mean *both* a “communication channel” *and* an “event,” the Examiner erases this distinction.

Nowhere does *Gupta* discuss a channel. However, in an effort to give the Examiner the benefit of the doubt, Appellant hypothesizes the existence of a data path between the connector 30 and the application collaboration module 40.

To the extent such a path exists, there is no suggestion that the application collaboration module 40 (or anything else) “subscribes” to that path. In fact, FIG. 2 of *Gupta* makes it quite clear that the application collaboration module 40 subscribes to an *event*, and not to some sort of data path.

*Gupta* clearly fails to disclose either publishing or subscribing to a *communication channel*. *Koo* fails to remedy this deficiency in the teaching of *Gupta*. Accordingly, the proposed combination of *Gupta* and *Koo*, even if it were possible, would fail to meet the limitations of step [3] and [4] in claim 1.

One of ordinary skill would surely have had the sense not to worry about overloading a data path when only two entities are able to use the path in the first place. Essentially, the



Examiner proposes to combine *Koo* with *Gupta* to solve a problem that one of ordinary skill would not even recognize in *Gupta*.

### **3.2 *Koo fails to disclose selecting a channel on the basis of a data object***

The Examiner concedes *Gupta*'s failure to disclose

*"the channel being selected on the basis of the data object"*

The Examiner proposes to remedy this deficiency by combining *Gupta* with *Koo*.

*Koo* discloses one channel carrying data from a first news service and another channel carrying data from a second news service. In particular, *Koo* teaches that:

*"one basic channel may carry data from a first news service, a second basic channel may carry data from a second news service, and a third basic channel may carry data from a third news service."*

According to the Examiner, the foregoing passage teaches choosing a channel on the basis of *the data object*.<sup>1</sup>

In fact, what the foregoing passage really teaches is choosing a channel on the basis of the news service, *not* the data object. According to this passage, if the three news services were to publish the same news story, three identical data objects would be sent on three different channels concurrently. Conversely, if a news service were to publish three different news stories, the three data objects representing those three news stories would be published on the same channel. This behavior is inconsistent with choosing the channel *on the basis of the data object*, as recited in step [3] of claim 1.

Accordingly, even if one were to somehow combine *Gupta* and *Koo*, the resulting combination would disclose a channel being selected on the basis of the *source* of a data object. This differs from claim 1 step [3], which recites a *"channel being selected on the basis of the data object."*

### **3.3 *Gupta fails to disclose channels subject to overloading***

The Examiner suggests that one of ordinary skill in the art would have combined *Koo* with *Gupta* to avoid overloading *Gupta*'s channels with too many subscribers.<sup>2</sup>

---

<sup>1</sup> *Final Office Action*, page 4, paragraph 9.

As a threshold matter, *Gupta* lacks any express disclosure of a communication channel that might possibly be overloaded by having too many subscribers. However, in an effort to give the Examiner the benefit of the doubt, Appellant hypothesizes the existence of a data path between the connector 30 and the application collaboration module 40.

To the extent such a path exists, it cannot be overloaded with subscribers. The disclosed architecture suggests that only two entities would use such a path: the connector 30, and the application collaboration module 40.

**3.4. *There is no motivation to combine Koo and Gupta***

As motivation to combine *Koo* with *Gupta*, the Examiner states that “connecting a large number of subscribers to a channel causes line loading and slows the transfer of data.”<sup>3</sup> However, this observation would prompt one of ordinary skill in the art to *limit* the number of subscribers on a channel, not to select a channel on the basis of the data object being published.

One of ordinary skill in the art would surely recognize that if latency arises from having too many subscribers, one should focus on reducing that number, and not on how the data object happened to find itself on that channel in the first place.

Accordingly, the proposed motivation to combine *Koo* and *Gupta* makes no technical sense.

**4. *The combination of Gupta, Koo and Bass fails to render obvious claims 12-15 and 18-48***

Independent claims 18 and 38 contain similar limitations. Accordingly, Appellant requests that arguments set forth below in connection with claim 18 be considered in connection with claim 38. Independent claims 18 and 28, and their progeny (claims 19-27 and claims 29-37) are being argued together as a group.

Claims 12-15 depend on claim 1, not claim 18. However, Appellant understands Rule 41.37 (c)(1)(vii) to require them to be argued in this section because they share the same ground of rejection as claims 18, 28 and progeny. However, only the arguments in subsections 4.4, 4.5,

---

<sup>2</sup> Final Office Action, page 4, paragraph 9.

<sup>3</sup> Final Office Action, page 4, paragraph 9.

and 4.8 are relevant to claims 12-15. In addition, claims 12-15 are patentable for at least the same reasons set forth in section 3.1-3.4 of this appeal brief.

To facilitate discussion, independent claim 18 is reproduced below with numbered paragraphs.

*18. A computer-implemented method of facilitating the exchange of information among applications, the method comprising:*

*[1] receiving a data object from a first application;*

*[2] using a first controller to route the received data object to a first transformer;*

*[3] using the first transformer to transform the data object from a first format used by the first application into a common format object;*

*[4] publishing the common format object to a communication channel;*

*[5] receiving a request from a subscribing application to subscribe to the communication channel;*

*[6] using a second controller to route the common format object to a second transformer;*

*[7] using the second transformer to transform the common format object into a data object in a second format used by the subscribing application; and*

*[8] sending the data object in the second format to the subscribing application.*

#### **4.1 Proposed routing step would serves no useful purpose in Gupta**

The Examiner concedes *Gupta*'s failure to disclose step [2] (the "routing step). To correct this deficiency in *Gupta*, the Examiner proposes that a broker 16, taken from *Bass*, be incorporated into the *Gupta* system. This broker 16 would act as a "first controller" that would then "route the received data object" back to the connector from which it came.

As best understood, the Examiner regards *Gupta* as carrying out step [1] when an application collaboration module 40 receives data from the interchange end of a connector 30. Thus, the “first application” recited in step [1] must be an application 70 connected to the application end of that connector 30.

Given this construction of step [1],

- the “received data object” of step [2] would have to be whatever data object was received by the application collaboration module 40 in step [1], and
- on the basis of step [3], the “first transformer” of step [2] would have to be the connector 30 that transformed the data object from a first format used by the application 70 to a “common format object” that can be processed by the application collaboration module 40.

One of ordinary skill in the art would no doubt have recognized that by the time step [1] is carried out, (i.e., by the time the module 40 receives data from a connector 30) the connector 30 (i.e. the “first transformer”) would already have processed that data. After all, *Gupta* FIG. 1 shows that connector 30 is *between* the application 70 and the module 40. Thus, one of ordinary skill in the art would have recognized no apparent purpose to adding step [2], which would only route the already transformed data object *back* to the connector 30 from which it came.

#### ***4.2 Proposed combination would solve an already-solved problem***

As motivation for introducing the *Bass* broker 16 to carry out step [2], the Examiner suggests that doing so would enable the various components in *Gupta* to anonymously publish and subscribe.<sup>4</sup>

However, one of ordinary skill in the art would quickly see that *Gupta* *already* discloses an event service 234 whose function is to provide the anonymous interaction that would allegedly be provided by incorporating the *Bass* broker 16.<sup>5</sup> Accordingly, one of ordinary skill in the art would have no reason to combine *Bass* with *Gupta* to solve a problem that has already been solved.

---

<sup>4</sup> *Final Office Action*, page 6, paragraph 24.

<sup>5</sup> *Gupta*, col. 7, lines 4-10.

#### **4.3 Communication channels in steps 4 and 5 are inconsistent**

The Examiner concedes *Gupta*'s failure to disclose step [4] ("publishing... to a communication channel"). To correct this deficiency in *Gupta*, the Examiner draws attention to *Koo*, which discloses publishing data on channels. Thus, the "communication channel" recited in step [4] would have to be like the channels from *Koo*'s FIG. 1.

The Examiner considers step [5] ("receiving a request... to subscribe") to be executed when an application collaboration module 40 subscribes to an *event*. Appellant therefore infers that the Examiner regards the "subscribing application" in step [5] to be the application collaboration module 40 and the "communication channel" in step [5] to be the event to which the application collaboration module 40 subscribes.

The Examiner's reading of the claim could only be consistent if *Koo*'s "channel" could reasonably be regarded as the same as *Gupta*'s "event." However, this does not appear possible. *Koo*'s "channel" is a path for carrying information. For example, *Koo* refers to architecture "especially adapted to allow information from a number of publishers to be placed on a single channel."<sup>6</sup> In contrast, the *Gupta* "event" is actual information that one might place on a channel. The *Gupta* "event" is thus completely different from the *Koo* "channel."

Although the Examiner is entitled to use the broadest reasonable meaning of claim terms, the idea that "channel" and "event" could be construed to mean the same thing is manifestly unreasonable.

#### **4.4 Motivation to combine *Koo* in claim 1 does not apply to claim 18**

In rejecting claim 1, the Examiner states that *Koo* discloses choosing a channel on the basis of the data object to be placed on a channel. The Examiner offered, as motivation for combining *Koo* with *Gupta*, the idea that doing so would prevent overloading of channels.

Claim 18 lacks the limitation of a "channel being selected on the basis of a data object." Therefore, the motivation to combine *Koo* with *Gupta* in claim 1 does not apply to claim 18. Accordingly, the section 103 rejection of claim 18 is improper because the Examiner has provided no basis for combining *Koo* with *Gupta* and *Bass*.

---

<sup>6</sup> *Koo*, col. 4, lines 44-47.

#### **4.5 No motivation to combine Bass with Gupta and Koo**

In rejecting claim 28, the Examiner concedes that the combination of *Gupta* and *Koo* fails to disclose “controller classes.” To remedy this deficiency, the Examiner proposes to transplant the brokers 16, 17 from *Bass* into a system made by combining *Gupta* and *Koo*. The brokers 16, 17 thus transplanted would play the role of “controller classes configured to route data objects to associated transformer classes.”

The Examiner suggests that one of ordinary skill in the art would have found it obvious to combine *Bass* with *Gupta* and *Koo* to enable anonymous publication and subscription. In doing so, the Examiner overlooks the fact one of ordinary skill would have promptly recognized that *Gupta*, by itself, *already* provided anonymous publication and subscription.

Appellant draws attention to FIG. 3 of *Gupta*, in which is disclosed an event service 234. The event service 234 “decouples information providers from information consumers”.<sup>7</sup> Accordingly, even without *Bass*, *Gupta already* provides anonymous publication and subscription. Clearly, one of ordinary skill in the art would not have been expected to combine *Bass* with *Gupta* and *Koo* to solve a problem that had already been solved by *Gupta* alone.

#### **4.6 Motivation to combine Koo with Gupta in claim 1 is unusable for claim 28**

The Examiner concedes that *Gupta* also fails to disclose a “shared object model configured to store data objects received from applications in a common format.” To supply this deficiency in the teaching of *Gupta*, the Examiner proposes to combine *Gupta* with *Koo*.

In connection with rejecting claim 1, the Examiner considered it obvious to combine *Gupta* with *Koo* because doing so would reduce latency caused by too many subscribers sharing the same channel.

Claim 28 is a different claim with limitations different from those in claim 1. In particular, claim 1 recited choosing a channel on the basis of a data object. Claim 28 does not even recite channels. Since claim 28 does not even include the limitation of choosing a channel on the basis of a data object and since it was this missing limitation that provided the basis for

---

<sup>7</sup> *Gupta*, col. 7, lines 4-10.

applying *Koo*, it hardly makes sense to reconstruct claim 28 by combining *Gupta* and *Koo* to reduce latency in communication channels.

**4.7 *Bass* fails to disclose any association between controllers and transformers**

Claim 28 recites “controller classes configured to route data objects to *associated* transformer classes.” Therefore, claim 28 requires a relationship between particular controllers and particular transformers. This relationship is described in paragraph 33 of the specification.

No such relationship exists between the broker 16 in *Bass* and the process adapters 18. For example, *Bass* FIG. 1 shows that the same broker 16 communicates with all process adapters 18. Nor does transplanting the *Bass* broker into *Gupta* suggest such a relationship. There is no suggestion in *Bass*, or in *Gupta*, that particular brokers (i.e. “controllers”) have *associated* connectors (i.e. “transformers”).

This is a distinction with a difference. Having transformers be associated with controllers means that adding and removing transformers does not require significant reprogramming of a complex entity. Instead, the process of adding and removing transformers can be done in a modular fashion.

**4.8. *Proposed motivation to combine Gupta and Koo amounts to hindsight reconstruction***

Applicant incorporates herein the argument made in connection with claim 1, as set forth in sections 3.3 and 3.4 of this brief.

**5. *Claims 38-48 recite patentable subject matter under 35 USC 101***

The Examiner states that independent claim 38 is non-statutory because the machine-readable medium is not limited to tangible embodiments. In doing so, the Examiner states that the claims be amended to recite only tangible computer-readable media and to avoid intangible transmission media.<sup>8</sup>

Appellant is puzzled by the maintenance of this rejection since the Response to the June 28, 2005 office action includes amendments along the lines the Examiner suggested. Appellant drew attention to this apparent oversight in a request for reconsideration filed on January 26, 2006. However, no advisory action was ever received.

---

<sup>8</sup> *Final Office Action*, page 2, paragraph 4.

Applicant : Lise Wiseman et al.  
Serial No. : 09/927,957  
Filed : August 9, 2001  
Page : 16 of 26

Attorney's Docket No.: 12587-008001 / 01315-00/US

As amended, claim 38 recites a "machine readable medium having encoded thereon instructions." Any reference to "a propagated signal" has been deleted. Accordingly, Appellant submits that the claim now recites tangible subject matter and is therefore within the scope of section 101's definition of statutory subject matter.

***Summary***

The brief fee in the amount of \$500 is being paid concurrently herewith on the Electronic Filing System (EFS) by way of Deposit Account authorization. Please apply all charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: May 16, 2006



Faustino A. Lichauco  
Reg. No. 41,942

Fish & Richardson P.C.  
225 Franklin Street  
Boston, MA 02110  
Telephone: (617) 542-5070  
Facsimile: (617) 542-8906



### **Appendix of Claims**

1. A computer-implemented method of exchanging information among applications, the method comprising:
  - providing a plurality of transformers, each transformer corresponding to a unique transformation from one format into another;
  - using a first transformer to transform a data object from a format understandable by a first application into a common format data object;
  - publishing the common format data object to a selected communication channel, the channel being selected on the basis of the data object;
  - subscribing to the communication channel to retrieve the published common format data object; and
  - using a second transformer to transform the common format data object into a format understandable by a second application.
2. The method of claim 1 wherein the data object corresponds to one or more of a plurality of business events.
3. The method of claim 1 wherein using the first transformer to transform the data object from the format understandable by the first application into the common format data object comprises translating the data object from a vendor-specific format associated with the first application to an Interface Data Language (IDL) object and storing the IDL object in a shared object model.
4. The method of claim 3 wherein the shared object model comprises a central repository of data objects corresponding to business events.

5. The method of claim 1 wherein using a first transformer to transform the data object from the format understandable by the first application into the common format data object is performed in response to a recognition of a business event by the first application.
6. The method of claim 1 wherein the method is performed in accordance with a plurality of process models that collectively define when information is to be exchanged among applications.
7. The method of claim 1 wherein publishing the common format data object to a communication channel is performed by a source connector and subscribing to the communication channel is performed by a target connector.
8. The method of claim 1 wherein publishing the common format data object to a communication channel is performed in accordance with a channel architecture that defines a plurality of communication channels having relative priorities.
9. The method of claim 1 wherein using the second transformer to transform the common format data object into the format understandable by the second application comprises retrieving a stored Interface Data Language. (IDL) format object from a central repository and translating the IDL object into a vendor-specific format associated with the second application.
10. The method of claim 1 in which information is exchanged among business support systems or operational support systems or a combination thereof.
11. The method of claim 1 in which at least one of the transformers comprises a class defined in an object-oriented programming language.
12. The method of claim 1 further comprising providing, for each transformer, a controller that is configured to route data objects to an associated transformer.
13. The method of claim 12, further comprising routing a data object to the first transformer using a first controller.

14. The method of claim 12, further comprising routing the common format data object to the second transformer using a second controller.
15. The method of claim 12 in which at least one of the controllers comprises a class defined in an object-oriented programming language.
16. The method of claim 1 further comprising using an acknowledgement class to exchange status messages among applications.
17. The method of claim 16 further comprising using the acknowledgement class to perform exception handling.
18. A computer-implemented method of facilitating the exchange of information among applications, the method comprising:
  - receiving a data object from a first application;
  - using a first controller to route the received data object to a first transformer;
  - using the first transformer to transform the data object from a first format used by the first application into a common format object;
  - publishing the common format object to a communication channel;
  - receiving a request from a subscribing application to subscribe to the communication channel;
  - using a second controller to route the common format object to a second transformer;
  - using the second transformer to transform the common format object into a data object in a second format used by the subscribing application; and

sending the data object in the second format to the subscribing application.

19. The method of claim 18 wherein the data object received from the first application corresponds to one or more of a plurality of business events.
20. The method of claim 18 wherein using the first transformer to transform the data object from the format used by the first application into the common format object comprises translating the data object from a vendor-specific format associated with the first application to an Interface Data Language. (IDL) object and storing the IDL object in a shared object model.
21. The method of claim 20 wherein the shared object model comprises a central repository of data objects corresponding to business events.
22. The method of claim 18 wherein using the first transformer to transform the data object from the format used by the first application into the common format object is performed in response to a recognition of a business event by the first application.
23. The method of claim 18 wherein the method is performed in accordance with a plurality of process models that collectively define when information is to be exchanged among applications.
24. The method of claim 18 wherein, if requests are received from a plurality of subscribing applications, then, for each subscribing application, the common format object is transformed using an associated transformer into a format corresponding to the subscribing application and sent to the subscribing application.
25. The method of claim 18 wherein publishing the common format data object to a communication channel is performed in accordance with a channel architecture that defines a plurality of communication channels having relative priorities.

26. The method of claim 18 wherein using the second transformer to transform the common format object into a data object in the second format used by the subscribing application comprises retrieving a stored Interface Data Language (IDL) format object from a central repository and translating the IDL object into a vendor-specific format associated with the subscribing application.
27. The method of claim 18 in which information is exchanged among business support systems or operational support systems or a combination thereof.
28. A system for facilitating the exchange of information among applications, the system comprising:

a plurality of digital computers, each of which executes an application, each application being configured to exchange information representative of business events with other applications; and

an integration hub in data communication with each of the digital computers for enabling transfer of information representative of business events between applications, the integration hub including a computer-readable medium on which encoded instructions for causing a computer to define

a plurality of process models each defining one or more conditions for sending a business event from an application to one or more other applications;

a shared object model configured to store data objects received from applications in a common format;

a plurality of transformer classes configured to translate data object from a format used by one or more applications into the common format or vice versa; and

a plurality of controller classes configured to route data objects to associated transformer classes.

29. The system of claim 28 further comprising a channel architecture defining a plurality of communication channels to which data objects from an application are to be published.
30. The system of claim 29 wherein the channel architecture defines relative priorities for the plurality of communication channels.
31. The system of claim 28 further comprising an acknowledgement class configured to exchange status messages among applications.
32. The system of claim 31 wherein the acknowledgement class is further configured to perform exception handling.
33. The system of claim 28 wherein each process model corresponds to a different business event.
34. The system of claim 28 wherein the shared object model comprises a central repository of data objects in an Interface Description Language. (IDL) format.
35. The system of claim 28 wherein each transformer class corresponds to a unique application format-common format translation.
36. The system of claim 28 wherein each controller class is configured to route data objects to an associated transformer class according to a process model.
37. The system of claim 28 wherein the transformer classes and the controller classes are implemented as classes in an object-oriented programming language.
38. A machine-readable medium having encoded thereon instructions for facilitating the exchange of information among applications, execution of the instructions causing one or more machines to perform operations comprising:

receiving a data object from a first application;

using a first controller to route the received data object to a first transformer;

using the first transformer to transform the data object from a first format used by the first application into a common format object;

publishing the common format object to a communication channel;

receiving a request from a subscribing application to subscribe to the communication channel;

using a second controller to route the common format object to a second transformer;

using the second transformer to transform the common format object into a data object in a second format used by the subscribing application; and

sending the data object in the second format to the subscribing application.

39. The instructions of claim 38 wherein the machine-readable instructions comprise computer software instructions executable by one or more computer systems.
40. The instructions of claim 38 wherein the data object received from the first application corresponds to one or more of a plurality of business events.
41. The instructions of claim 38 wherein using the first transformer to transform the data object from the format used by the first application into the common format object comprises translating the data object from a vendor-specific format associated with the first application to an Interface Data Language. (IDL) object and storing the IDL object in a shared object model.
42. The instructions of claim 41 wherein the shared object model comprises a central repository of data objects corresponding to business events.

43. The instructions of claim 38 wherein using the first transformer to transform the data object from the format used by the first application into the common format object is performed in response to a recognition of a business event by the first application.
44. The instructions of claim 38 wherein one or more of the instructions are executed in accordance with a plurality of process models that collectively define when information is to be exchanged among applications.
45. The instructions of claim 38 wherein, if requests are received from a plurality of subscribing applications, then, for each subscribing application, the common format object is transformed using an associated transformer into a format corresponding to the subscribing application and sent to the subscribing application.
46. The instructions of claim 38 wherein publishing the common format data object to a communication channel is performed in accordance with a channel architecture that defines a plurality of communication channels having relative priorities.
47. The instructions of claim 38 wherein using the second transformer to transform the common format object into the data object in the second format used by the subscribing application comprises retrieving a stored Interface Data Language (IDL) format object from a central repository and translating the IDL object into a vendor-specific format associated with the subscribing application.
48. The instructions of claim 38 in which information is exchanged among business support systems or operational support systems or a combination thereof.



Applicant : Lise Wiseman et al.  
Serial No. : 09/927,957  
Filed : August 9, 2001  
Page : 25 of 26

Attorney's Docket No.: 12587-008001 / 01315-00/US

### **Evidence Appendix**

None

Applicant : Lise Wiseman et al.  
Serial No. : 09/927,957  
Filed : August 9, 2001  
Page : 26 of 26

Attorney's Docket No.: 12587-008001 / 01315-00/US

**Related Proceedings Appendix**

None